

#2
2 2602

S/N unknown

PATENT

IN THE UNITED STATES PATENT AND TRADEMARK OFFICE

Applicant: YOSHIOKA Serial No.: unknown
Filed: concurrent herewith Docket No.: 10873.810US
Title: A MICROPROCESSOR AND A COMPILER

JC979 U.S. PTO
10/033446
12/27/01

CERTIFICATE UNDER 37 CFR 1.10

'Express Mail' mailing label number: EV 037641170 US

Date of Deposit: December 27, 2001

I hereby certify that this correspondence is being deposited with the United States Postal Service 'Express Mail Post Office To Addressee' service under 37 CFR 1.10 on the date indicated above and is addressed to the Assistant Commissioner for Patents, Washington, D.C. 20231.

By:

Name: Chris Stordahl

SUBMISSION OF PRIORITY DOCUMENT

Assistant Commissioner for Patents
Washington, D.C. 20231

Dear Sir:

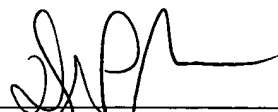
Applicants enclose herewith one certified copy of a Japanese application, Serial No. 2000-403540, filed December 28, 2000, the right of priority of which is claimed under 35 U.S.C. § 119.

Respectfully submitted,

MERCHANT & GOULD P.C.
P.O. Box 2903
Minneapolis, Minnesota 55402-0903
(612) 332-5300

Dated: December 27, 2000

By:



Douglas P. Mueller
Reg. No. 30,300

DPM/tvm

日本国特許庁
JAPAN PATENT OFFICE

JC979 U.S. PTO
10/033446
12/27/01

別紙添付の書類に記載されている事項は下記の出願書類に記載されている事項と同一であることを証明する。

This is to certify that the annexed is a true copy of the following application as filed with this Office

出願年月日

Date of Application:

2000年12月28日

出願番号

Application Number:

特願2000-403540

出願人

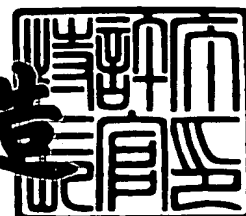
Applicant(s):

松下電器産業株式会社

2001年 8月24日

特許庁長官
Commissioner,
Japan Patent Office

及川耕造



出証番号 出証特2001-3075487

【書類名】 特許願

【整理番号】 R4458

【提出日】 平成12年12月28日

【あて先】 特許庁長官 殿

【国際特許分類】 G06F 9/26
G06F 9/320

【発明者】

【住所又は居所】 大阪府門真市大字門真 1 0 0 6 番地 松下電器産業株式会社内

【氏名】 吉岡 志郎

【特許出願人】

【識別番号】 000005821

【氏名又は名称】 松下電器産業株式会社

【代理人】

【識別番号】 100095555

【弁理士】

【氏名又は名称】 池内 寛幸

【電話番号】 06-6361-9334

【選任した代理人】

【識別番号】 100076576

【弁理士】

【氏名又は名称】 佐藤 公博

【選任した代理人】

【識別番号】 100107641

【弁理士】

【氏名又は名称】 鎌田 耕一

【選任した代理人】

【識別番号】 100110397

【弁理士】

【氏名又は名称】 帋丘 圭司

【選任した代理人】

【識別番号】 100115255

【弁理士】

【氏名又は名称】 辻丸 光一郎

【選任した代理人】

【識別番号】 100115152

【弁理士】

【氏名又は名称】 黒田 茂

【手数料の表示】

【予納台帳番号】 012162

【納付金額】 21,000円

【提出物件の目録】

【物件名】 明細書 1

【物件名】 図面 1

【物件名】 要約書 1

【包括委任状番号】 0004605

【プルーフの要否】 要

【書類名】 明細書

【発明の名称】 マイクロプロセッサ及び命令列変換装置

【特許請求の範囲】

【請求項 1】 分岐が成立する分岐予測を行うマイクロプロセッサにおいて、
分岐が成立しなかった場合に次に実行される命令が制限されている制限付条件
分岐命令を備え、

命令列のデコードにおいて前記制限付条件分岐命令をデコードし、分岐が成立
しないことを検出した場合、前記次に実行される命令を、通常命令のフェッチと
デコードに要するサイクル数よりも少ないサイクル数でフェッチとデコードを行
うことを特徴とするマイクロプロセッサ。

【請求項 2】 分岐が成立する分岐予測を行うマイクロプロセッサにおいて、
分岐が成立しなかった場合に次に実行される命令が制限されている制限付条件
分岐命令と、

命令列を格納する記憶装置と、

前記分岐が成立しなかった場合に次に実行される命令のオペコードを格納する
特殊レジスタとを備え、

命令列のデコードにおいて前記制限付条件分岐命令をデコードし、分岐しない
ことを検出したとき、オペコードを前記特殊レジスタから、オペランドを前記記
憶装置からデコーダへ供給することを特徴とするマイクロプロセッサ。

【請求項 3】 分岐が成立する分岐予測を行うマイクロプロセッサにおいて、
分岐が成立しなかった場合に次に実行される命令が制限されている制限付条件
分岐命令と、

通常の命令列のデコードに用いる第 1 のデコーダと、

前記制限付条件分岐命令において分岐が成立しなかった場合に次に実行される
命令のみを通常のデコードサイクル数よりも短いサイクル数でデコードする第 2
のデコーダとを備え、

前記第 1 のデコーダが前記制限付条件分岐命令をデコードし、分岐しないこと
を検出したときは、前記次に実行される命令のデコードサイクルにおいて前記第
2 のデコーダを用いることを特徴とするマイクロプロセッサ。

【請求項 4】 分岐が成立しなかった場合に次に実行できる命令が制限されている命令を制限付条件分岐命令とし、

入力された命令列において条件分岐命令を検出した場合、当該条件分岐が成立しなかった場合に次に実行される命令を調べ、当該条件分岐命令と分岐が成立しなかった場合に次に実行される命令との関係が、前記制限付き条件分岐命令と分岐が成立しなかった場合に次に実行できる命令との関係である場合、前記条件分岐命令を前記制限付条件分岐命令で置き換えることを特徴とする命令列変換装置。

【請求項 5】 分岐が成立しない分岐予測を行うマイクロプロセッサにおいて

分岐が成立した場合に次に実行される分岐先命令が制限されている制限付条件分岐命令を備え、

命令列のデコードにおいて前記制限付条件分岐命令をデコードし、分岐が成立したことを検出した場合、前記次に実行される分岐先命令を、通常命令のフェッチとデコードに要するサイクル数よりも少ないサイクル数でフェッチとデコードを行うことを特徴とするマイクロプロセッサ。

【請求項 6】 分岐が成立しない分岐予測を行うマイクロプロセッサにおいて

分岐が成立した場合に次に実行される分岐先命令が制限されている制限付条件分岐命令と、

命令列を格納する記憶装置と、

命令列のデコードにおいて前記制限付条件分岐命令をデコードし、分岐することを検出したとき、分岐先命令のオペコードを前記特殊レジスタから、オペランドを前記記憶装置からデコーダへ供給することを特徴とするマイクロプロセッサ。

【請求項 7】 分岐が成立しない分岐予測を行うマイクロプロセッサにおいて

分岐が成立した場合に次に実行される分岐先命令が制限されている制限付条件分岐命令と、

通常の命令列のデコードに用いる第 1 のデコーダと、

前記制限付条件分岐命令において分岐が成立した場合に次に実行される分岐先命令のみを通常のデコードサイクル数よりも短いサイクル数でデコードする第 2 のデコーダとを備え、

前記第 1 のデコーダが前記制限付条件分岐命令をデコードし、分岐することを検出したときは、前記次に実行される分岐先命令のデコードサイクルにおいて前記第 2 のデコーダを用いることを特徴とするマイクロプロセッサ。

【請求項 8】 分岐が成立した場合に次に実行できる命令が制限されている命令を制限付条件分岐命令とし、

入力された命令列において条件分岐命令を検出した場合、当該条件分岐が成立した場合に次に実行される分岐先命令を調べ、当該条件分岐命令と分岐先命令との関係が、前記制限付条件分岐命令と分岐先命令との関係である場合、前記条件分岐命令を前記制限付条件分岐命令で置き換えることを特徴とする命令列変換装置。

【発明の詳細な説明】

【 0 0 0 1 】

【発明の属する技術分野】

本発明は、条件分岐命令を高速に実行するマイクロプロセッサおよび命令列変換装置に関する。

【 0 0 0 2 】

【従来の技術】

従来技術において分岐のペナルティを軽減する手法として、分岐予測と遅延分岐とが知られている。

【 0 0 0 3 】

図 1 1 に従来技術における分岐予測機能を備えたマイクロプロセッサの動作を示す。以下の例では、分岐するとの予測のもとにペナルティが小さくなるように処理の流れを制御したものを例として説明する。

【 0 0 0 4 】

図 1 1 (a) および図 1 1 (b) において、縦軸に現われる I F , D E C , E

X, ME, WBはそれぞれ5段パイプラインのそれぞれのステージを意味し、IFは命令フェッチ、DECはデコード、EXは実行、MEはメモリアクセス、WBはライトバックである。また、全部で4つもしくは5つのサイクルをあらわしている。

【0005】

図11(a)は分岐が成立する場合、つまり、分岐予測がヒットした場合の処理を示している。分岐が成立した場合は、分岐先命令を格納したターゲットレジスタTRから即座に命令を供給するためペナルティは発生せず、EXステージでは分岐命令実行後、続いて分岐先の命令が実行されている。

【0006】

しかし、分岐が成立しなかった場合、つまり、分岐予測がヒットしなかった場合はペナルティが発生する。図11(b)は分岐が成立しなかった場合の処理を示している。分岐が成立しなかった場合は、命令シーケンス上において分岐命令の次に置かれている命令を実行することとなり、当該次の命令のフェッチから始めるため、1サイクル分のペナルティが発生することとなる。

【0007】

このように分岐予測機能は、ループのように分岐が殆ど成立し、分岐予測がヒットする割合が高い処理においてはトータルでのペナルティが小さくなり有利である。

【0008】

次に、図12に従来技術における遅延分岐機能を備えたマイクロプロセッサの動作を示す。図12(a)は分岐が成立する場合、図12(b)は分岐が成立しない場合を示している。遅延分岐機能は、パイプラインにおいて、分岐命令の後に遅延スロットを設けておき、分岐命令に続いて遅延スロットを処理することによりパイプラインの流れを保証しつつ、分岐命令の分岐判断を見てからフェッチする命令を決めるものである。分岐命令の分岐判断に従って分岐先の命令か、命令シーケンス上において分岐命令の次に置かれている命令のいずれかをフェッチする。遅延スロットには分岐結果に影響を受けず、先行して処理しうる命令があれば当該命令を遅延スロットに割り当てて先に実行し、そのような命令がない場

合にはNOP（無効命令）を遅延スロットに割り当てる必要がある。

【0009】

遅延分岐機能を用いる場合は、分岐判断の結果によらずいずれの場合も遅延スロット分を待ち合わせる事となり、遅延スロットにNOPを割り当てた場合には結局遅延スロットのサイクル分がペナルティとなる。

【0010】

【発明が解決しようとする課題】

図11の分岐予測機能を備えた従来技術におけるマイクロプロセッサでは、複数の分岐命令を想定した場合、ターゲットレジスタを複数設ける必要があり、ハードウェアが大きくなるという問題がある。また、分岐先を先行フェッチしておけばターゲットレジスタを省略できるが、分岐が成立しなかった場合は命令のフェッチをやり直すため、ペナルティが発生する。分岐する確率が50%のときペナルティは最大となる。

【0011】

また、遅延分岐機能を備えたマイクロプロセッサでは、遅延スロットに命令を置けない場合はNOPを割り当てるため、この場合、分岐するしないにかかわらず常に遅延スロット分のペナルティが発生してしまうという問題がある。

【0012】

家電製品の制御においては、スイッチ入力や機器の状態や外部入力を制御するためcase文による分岐命令の処理割合が多い。図13にcase文の例を示す。図13（a）がcase文による記述であり、図13（b）はそれをコンパイルした結果のアセンブラ記述である。ここで、ldはロード命令、cmpは比較命令、jzは条件分岐命令を示している。case文による分岐はループと違い、分岐が成立する確率が低い。つまり、従来技術ではペナルティが発生する確率が高くなり、トータルとしてペナルティが大きくなってしまふ。

【0013】

上記のようにペナルティが発生すると、マイクロプロセッサの処理効率が低下し、その結果、消費電力の増大にもつながってしまう。

【0014】

本発明はかかる課題に鑑み、条件分岐命令の実行により分岐予測が外れた場合でもペナルティを発生しないようにしてマイクロプロセッサの処理効率低下を防止し、低消費電力を達成することを目的とする。

【 0 0 1 5 】

【課題を解決するための手段】

上記課題を解決するため本発明のマイクロプロセッサは、分岐が成立するとの分岐予測を行うマイクロプロセッサにおいて、分岐が成立しなかった場合に次に実行される命令が制限されている制限付条件分岐命令を備え、命令列のデコードにおいて前記制限付条件分岐命令をデコードし、分岐が成立しないことを検出した場合、前記次に実行される命令を、通常命令のフェッチとデコードに要するサイクル数よりも少ないサイクル数でフェッチとデコードを行うことを特徴とする。

【 0 0 1 6 】

上記構成により、分岐が成立するとの分岐予測を行うマイクロプロセッサにおいて、分岐予測が外れた場合でも、次に実行される命令が制限されているため、通常命令のフェッチとデコードに要するサイクル数よりも少ないサイクル数でフェッチとデコードを行って実行ステージに渡すことができ、ペナルティ発生を防止することができ、処理効率を向上させ、低消費電力化を達成できる。

【 0 0 1 7 】

次に、上記課題を解決するため本発明のマイクロプロセッサは、分岐が成立するとの分岐予測を行うマイクロプロセッサにおいて、分岐が成立しなかった場合に次に実行される命令が制限されている制限付条件分岐命令と、命令列を格納する記憶装置と、前記分岐が成立しなかった場合に次に実行される命令のオペコードを格納する特殊レジスタとを備え、命令列のデコードにおいて前記制限付条件分岐命令をデコードし、分岐しないことを検出したとき、オペコードを前記特殊レジスタから、オペランドを前記記憶装置からデコーダへ供給することを特徴とする。

【 0 0 1 8 】

上記構成により、分岐が成立するとの分岐予測を行うマイクロプロセッサにお

いて、分岐予測が外れた場合でも、次に実行される命令が制限され、当該命令のオペコードが特殊レジスタから、当該命令のオペランドが記憶装置からデコーダへ即座に供給され、デコード結果を実行ステージに渡すことができるので、ペナルティの発生を防止することができ、処理効率を向上させ、低消費電力化を達成できる。

【 0 0 1 9 】

次に、上記課題を解決するため本発明のマイクロプロセッサは、分岐が成立するとの分岐予測を行うマイクロプロセッサにおいて、分岐が成立しなかった場合に次に実行される命令が制限されている制限付条件分岐命令と、通常の命令列のデコードに用いる第1のデコーダと、前記制限付条件分岐命令において分岐が成立しなかった場合に次に実行される命令のみを通常のデコードサイクル数よりも短いサイクル数でデコードする第2のデコーダとを備え、前記第1のデコーダが前記制限付条件分岐命令をデコードし、分岐しないことを検出したときは、前記次に実行される命令のデコードサイクルにおいて前記第2のデコーダを用いることを特徴とする。

【 0 0 2 0 】

上記構成により、分岐が成立するとの分岐予測を行うマイクロプロセッサにおいて、分岐予測が外れた場合でも、次に実行される命令が制限され、当該命令のデコードに特化した第2のデコードを用いることにより即座にデコードを実行し、デコード結果を実行ステージに渡すことができるので、ペナルティの発生を防止することができ、処理効率を向上させ、低消費電力化を達成できる。

【 0 0 2 1 】

次に、上記課題を解決するため本発明の命令列変換装置は、分岐が成立しなかった場合に次に実行できる命令が制限されている命令を制限付条件分岐命令とし、入力された命令列において条件分岐命令を検出した場合、当該条件分岐が成立しなかった場合に次に実行される命令を調べ、当該条件分岐命令と分岐が成立しなかった場合に次に実行される命令との関係が、前記制限付き条件分岐命令と分岐が成立しなかった場合に次に実行できる命令との関係である場合、前記条件分岐命令を前記制限付条件分岐命令で置き換えることを特徴とする。

【 0 0 2 2 】

上記構成により、本発明の命令列変換装置を用いれば、従来技術において用いられる通常のコンパイラによるプログラムのコンパイル結果を入力とし、当該コンパイル結果に含まれている条件分岐命令のうち、制限付条件分岐命令に変換することができる命令列を制限付条件分岐命令列に置き換えて出力することができ、本発明の制限付き条件分岐命令を処理できるマイクロプロセッサに用いるプログラムを得ることができる。

【 0 0 2 3 】

次に、上記課題を解決するため本発明のマイクロプロセッサは、分岐が成立しないとの分岐予測を行うマイクロプロセッサにおいて、分岐が成立した場合に次に実行される分岐先命令が制限されている制限付条件分岐命令を備え、命令列のデコードにおいて前記制限付条件分岐命令をデコードし、分岐が成立したことを検出した場合、前記分岐先命令を通常命令のフェッチとデコードに要するサイクル数よりも少ないサイクル数でフェッチとデコードすることを特徴とする。

【 0 0 2 4 】

上記構成により、分岐が成立しないとの分岐予測を行うマイクロプロセッサにおいて、分岐予測が外れた場合でも次に実行される分岐先命令が制限されているため、通常命令のフェッチとデコードに要するサイクル数よりも少ないサイクル数で分岐先命令のフェッチとデコードを行って実行ステージに渡すことができ、ペナルティ発生を防止することができ、処理効率を向上させ、低消費電力化を達成できる。

【 0 0 2 5 】

次に、上記課題を解決するため本発明のマイクロプロセッサは、分岐が成立しないとの分岐予測を行うマイクロプロセッサにおいて、分岐が成立した場合に次に実行される分岐先命令が制限されている制限付条件分岐命令と、命令列を格納する記憶装置と、命令列のデコードにおいて前記制限付条件分岐命令をデコードし、分岐することを出したとき、前記分岐先命令のオペコードを前記特殊レジスタから、オペランドを前記記憶装置からデコーダへ供給することを特徴とする。

【 0 0 2 6 】

上記構成により、分岐が成立しないとの分岐予測を行うマイクロプロセッサにおいて、分岐予測が外れた場合でも、次に実行される分岐先命令が制限され、当該分岐先命令のオペコードが特殊レジスタから、当該分岐先命令のオペランドが記憶装置からデコーダへ即座に供給され、デコード結果を実行ステージに渡すことができるので、ペナルティの発生を防止することができ、処理効率を向上させ、低消費電力化を達成できる。

【 0 0 2 7 】

次に、上記課題を解決するため本発明のマイクロプロセッサは、分岐が成立しないとの分岐予測を行うマイクロプロセッサにおいて、分岐が成立した場合に次に実行される分岐先命令が制限されている制限付条件分岐命令と、通常の命令列のデコードに用いる第1のデコーダと、前記制限付条件分岐命令において分岐が成立した場合に分岐先命令のみを通常のデコードサイクル数よりも短いサイクル数でデコードする第2のデコーダとを備え、前記第1のデコーダが前記制限付条件分岐命令をデコードし、分岐することを検出したときは、前記分岐先命令のデコードサイクルにおいて前記第2のデコーダを用いることを特徴とする。

【 0 0 2 8 】

上記構成により、分岐が成立しないとの分岐予測を行うマイクロプロセッサにおいて、分岐予測が外れた場合でも、次に実行される分岐先命令が制限され、当該命令のデコードに特化した第2のデコードを用いることにより即座にデコードを実行し、デコード結果を実行ステージに渡すことができるので、ペナルティの発生を防止することができ、処理効率を向上させ、低消費電力化を達成できる。

【 0 0 2 9 】

次に、上記課題を解決するため本発明の命令列変換装置は、分岐が成立した場合に次に実行できる命令が制限されている命令を制限付条件分岐命令とし、入力された命令列において条件分岐命令を検出した場合、当該条件分岐が成立した場合に次に実行される命令を調べ、当該条件分岐命令と分岐が成立した場合に次に実行される命令との関係が、前記制限付き条件分岐命令と分岐が成立した場合に次に実行できる命令との関係である場合、前記条件分岐命令を前記制限付条件分

岐命令で置き換えることを特徴とする。

【 0 0 3 0 】

上記構成により、本発明の命令列変換装置を用いれば、従来技術における通常のコンパイラによるプログラムのコンパイル結果を入力とし、当該コンパイル結果に含まれている条件分岐命令のうち、制限付条件分岐命令に変換することができ、命令列を制限付条件分岐命令列に置き換えて出力することができ、本発明の制限付き条件分岐命令を処理できるマイクロプロセッサに用いるプログラムを得ることができる。

【 0 0 3 1 】

【発明の実施の形態】

(実施形態 1)

以下、本発明の実施形態 1 のマイクロプロセッサを説明する。

【 0 0 3 2 】

実施形態 1 のマイクロプロセッサは、分岐が成立するとの分岐予測を行い、分岐が成立しなかった場合でもペナルティ発生を防止するものである。

【 0 0 3 3 】

case 文などで書かれたプログラムにおいて分岐命令はループと異なり、分岐が成立する確立は高くない。また、分岐命令の直後に比較命令が置かれることが多い。本発明のマイクロプロセッサは制限付条件分岐命令を用いている。ここで、制限付条件分岐命令とは、条件分岐命令の一種であり、分岐が成立しなかった場合に次に実行される命令が制限されているという命令である。制限付条件分岐命令のデコードの結果、分岐が成立しないことを検出した場合、次に実行される命令を通常命令のフェッチとデコードに要するサイクル数よりも少ないサイクル数でフェッチとデコードを行うことによりペナルティ発生を防止する。

【 0 0 3 4 】

本実施形態 1 では、次に実行される命令を少ないサイクル数でフェッチとデコードを行う手段として、命令列を格納する記憶装置と分岐が成立しなかった場合に次に実行される命令のオペコードを格納する特殊レジスタとを備え、制限付条件分岐命令をデコードした結果、分岐しないことを検出したとき、オペコードを

特殊レジスタから、オペランドを記憶装置からデコーダへ供給するという手段を用いる。

【 0 0 3 5 】

なお、分岐するとの分岐予測が外れた場合に次に実行される命令とは、通常は命令シーケンス上で制限付条件分岐命令の次に置かれている命令であるので、以下の説明では、制限付条件分岐命令の次に置かれている命令として説明する。

【 0 0 3 6 】

図 1 は、上記処理を実現する構成例として命令フェッチ部と記憶装置と命令レジスタを模式的に示した構成図である。

【 0 0 3 7 】

図 1 において、1 0 0 は命令列が格納された記憶装置、1 0 1 は制限付条件分岐命令を実行し分岐が成立しなかった場合に制限付条件分岐命令の次に置かれている命令のオペコードを格納した特殊レジスタ、1 0 2 は記憶装置に置かれた命令をフェッチするフェッチ部、1 0 3 はフェッチ部 1 0 2 の値をフェッチするフェッチ部、1 0 5 は記憶装置 1 0 0、フェッチ部 1 0 2、フェッチ部 1 0 3、特殊レジスタ 1 0 1 の値を選択して出力する選択部、1 0 6 はどれを選択するかを示す選択制御信号、1 0 4 は選択部の出力を格納してデコーダへその値を出力する命令レジスタである。

【 0 0 3 8 】

特殊レジスタ 1 0 1 には制限付条件分岐命令の次に置かれた命令のオペコードを格納しておく。例えば、制限された命令種類として比較命令 `cmp` などが特殊レジスタ 1 0 1 に格納される。このように予め決まったオペコードを用意しておくので、制限付条件分岐命令の次に置かれた命令については即座にオペコードが供給できる構成になっている。

【 0 0 3 9 】

図 1 の構成において、記憶装置 1 0 0 に格納されている命令は、フェッチ部 1 0 2 とフェッチ部 1 0 3 により二段階でフェッチされている。ここで、選択部 1 0 5 は、選択制御信号 1 0 6 に従って記憶装置 1 0 0、フェッチ部 1 0 2、フェッチ部 1 0 3、特殊レジスタ 1 0 1 の値を選択して命令レジスタ 1 0 4 に渡すの

で、命令レジスタ 1 0 4 は、4 つの値を選択して受け取ることができる。つまり、制限付条件分岐命令を実行し、分岐予測が外れ、分岐が成立しない場合でも、予め用意されたオペコードを特殊レジスタ 1 0 1 から選択部 1 0 5 を介して命令レジスタ 1 0 4 へ供給し、また、オペランドを記憶装置 1 0 0 から選択部 1 0 5 を介して命令レジスタ 1 0 4 へ供給することで、制限付条件分岐命令の次に置かれた命令のフェッチとデコードとを 1 サイクルで行うことができる。一般にデコード時間はオペコードのデコードが支配的であり、オペランドはレジスタを選択するためにデコードされるだけなので時間がかからない。それゆえ、特殊レジスタ 1 0 1 からオペコードを供給することで早くデコードが開始され、1 サイクルでデコードを完了することができる。

【 0 0 4 0 】

図 2 は、図 1 に示した本発明の実施形態 1 の構成の動作を示したタイミングチャートである。

【 0 0 4 1 】

図 2 において、縦軸の I F, D E C, E X, M E, W B はそれぞれ 5 段パイプラインのそれぞれのステージを意味し、I F は命令フェッチ、D E C はデコード、E X は実行、M E はメモリアクセス、W B はライトバックである。また、全部で 4 つのサイクルをあらわしている。また、図中、W B の下には命令レジスタ 1 0 4 に供給されるオペコード部とオペランド部の供給元を示した。

【 0 0 4 2 】

図 2 (a) は分岐予測が当たり、分岐が成立する場合の動作である。

【 0 0 4 3 】

サイクル 1 で制限付条件分岐命令がフェッチされる。ここで制限付条件分岐命令はフェッチ部 1 0 3 に格納されている。さらに制限付条件分岐命令がプリデコードされ、分岐先アドレスが計算される。次に、サイクル 2 では、分岐するとの予測のもと、分岐先の命令がフェッチ部 1 0 2 にフェッチされる。また、D E C ステージでは、制限付条件分岐命令がデコードされ、分岐が成立することが検知される。次に、サイクル 3 では、分岐先命令の次の命令がフェッチ部 1 0 2 にフェッチされ、分岐先の命令は D E C ステージにおいてデコードされる。このよう

に分岐するとの分岐予測が当たった場合には分岐処理に伴うペナルティは発生していない。

【 0 0 4 4 】

図 2 (b) は分岐が成立しない場合の動作である。

【 0 0 4 5 】

サイクル 1 で制限付条件分岐命令がフェッチされる。ここで制限付条件分岐命令はフェッチ部 1 0 3 に格納されている。また、制限付条件分岐命令がプリデコードされ、分岐先アドレスが計算される。

【 0 0 4 6 】

次に、サイクル 2 では、分岐するとの予測のもと、フェッチ部 1 0 2 には分岐先の命令がフェッチ部 1 0 2 にフェッチされる。一方、DEC では、制限付条件分岐命令がデコードされ、分岐が成立しないことが検知される。

【 0 0 4 7 】

次に、サイクル 3 では、オペコードとして特殊レジスタ 1 0 1 から供給され、オペランドとして記憶装置 1 0 0 から読み出された値が命令レジスタ 1 0 4 に供給される。ここで、制限付条件分岐命令の次に置かれている命令のオペコードは特殊レジスタ 1 0 1 に格納されたオペコードと一致するように命令の種類が制限されているため、特殊レジスタ 1 0 1 からオペコードを命令レジスタ 1 0 4 に供給することで制限付条件分岐命令の次に置かれている命令と同等のオペコードのデコードが早く開始できる。また、オペランドのデコードはレジスタを選択するためだけのものなので、記憶装置 1 0 0 から供給してもサイクル 3 内でデコードできる。

【 0 0 4 8 】

なお、サイクル 3 のフェッチステージでは、記憶装置 1 0 0 から制限付条件分岐命令のオペランドを読み出すと同時にその次に配置された命令がフェッチ部 1 0 2 にフェッチされる。

【 0 0 4 9 】

次に、サイクル 4 では、制限付条件分岐命令の次の次に置かれている命令がフェッチ部 1 0 2 にフェッチされ、制限付条件分岐命令の次に置かれている命令が

デコードされることとなる。

【 0 0 5 0 】

このように、本実施形態 1 のマイクロプロセッサでは、分岐するとの分岐予測が外れた場合においても、分岐処理に伴うペナルティは発生しない。

【 0 0 5 1 】

以上、本発明の実施形態 1 にかかるマイクロプロセッサによれば、分岐が成立するとの分岐予測を行うマイクロプロセッサにおいて、分岐予測が外れた場合でも、制限付条件分岐命令の次の命令のオペコードが特殊レジスタから、当該命令のオペランドが記憶装置からデコーダへ即座に供給され、デコード結果を実行ステージに渡すことができるので、ペナルティの発生を防止することができ、処理効率を向上させ、低消費電力化を達成できる。

【 0 0 5 2 】

(実施形態 2)

以下、本発明の実施形態 2 のマイクロプロセッサを説明する。

【 0 0 5 3 】

実施形態 2 のマイクロプロセッサも、実施形態 1 と同様、分岐が成立するとの分岐予測を行い、分岐が成立しなかった場合でもペナルティ発生を防止するものである。また、制限付条件分岐命令を用いる点においても実施形態 1 と同様である。

【 0 0 5 4 】

本実施形態 2 では、分岐予測が外れ、制限付条件分岐命令の次に置かれた命令を少ないサイクル数でフェッチとデコードを行う手段として、通常の命令列のデコードに用いる第 1 のデコーダと、制限付条件分岐命令の次に置かれた命令のデコードに特化してその命令を通常のデコードサイクル数よりも短いサイクル数でデコードする第 2 のデコーダを用いる。通常、第 1 のデコーダを用い、第 1 のデコーダが制限付条件分岐命令をデコードし、分岐しないことを検出したときは、制限付条件分岐命令の次に置かれた命令のデコードサイクルにおいて第 2 のデコーダを用いて短時間でデコードを実行するものである。

【 0 0 5 5 】

図 3 は、上記処理を実現する構成例として、デコーダを中心とした構成を模式的に示した図である。

【 0 0 5 6 】

図 3 において、3 0 0 は命令レジスタ、3 0 1 は D E C ステージで通常メインに用いられる第 1 のデコーダ、3 0 2 は分岐予測が外れた場合に用いられる第 2 のデコーダ、3 0 3 はデコーダ選択信号、3 0 4 は演算器等を制御する制御信号、3 0 5 は第 1 のデコーダと第 2 のデコーダとを選択する選択部、3 0 6 は選択部 3 0 5 にどちらのデコーダを選択するかを指示する選択制御信号、3 0 7 は第 1 のデコーダの制御信号と第 2 のデコーダの制御信号とを選択する選択部、3 0 8 は選択部 3 0 5 にどちらのデコーダの制御信号を選択するかを指示する選択制御信号である。

【 0 0 5 7 】

本実施形態 2 では、第 1 のデコーダが D E C ステージで通常メインに用いられる構成となっている。さらに、制限付条件分岐命令の次に置かれている命令のデコードを行う特化された専用の第 2 のデコーダを設けた構成となっている。例えば、制限された命令種類として比較命令 c m p などが第 2 のデコーダでデコードされる命令となる。第 2 のデコーダは非常に限られた命令のデコードに特化した専用モジュールであるので、ハードウェア規模が小さくデコードの時間が短い。このため、分岐予測が外れた場合であっても、制限付条件分岐命令の実行ステージのサイクル内で、制限付条件分岐命令の次に置かれた命令のフェッチとデコードとを行うことができ、ペナルティの発生を防止することができる。

【 0 0 5 8 】

図 4 は、図 3 に示した本発明の実施形態 2 の構成の動作を示したタイミングチャートである。

【 0 0 5 9 】

図 4 も図 2 と同様、縦軸の I F, D E C, E X, M E, W B はそれぞれ 5 段パイプラインのそれぞれのステージを意味し、I F は命令フェッチ、D E C はデコード、E X は実行、M E はメモリアクセス、W B はライトバックである。また、全部で 4 つのサイクルをあらわしている。また、図 4 (b) では、W B の下に用

いられるデコーダを示した。

【0060】

図4（a）は分岐予測が当たり、分岐が成立する場合の動作である。ここではデコードは全て第1のデコーダ301で行う。まず、サイクル1で制限付条件分岐命令がフェッチされる。ここで制限付条件分岐命令はフェッチ部103に格納されている。さらにサイクル1では制限付条件分岐命令をプリデコードして分岐先アドレスを計算する。次に、サイクル2ではフェッチ部には分岐するとの予測のもと、分岐先の命令がフェッチ部102にフェッチされる。一方、第1のデコーダ301では、制限付条件分岐命令がデコードされ、分岐が成立することが検知される。次に、サイクル3では分岐先命令の次の命令がフェッチ部102にフェッチされ、分岐先の命令は第1のデコーダ301においてデコードされる。このように分岐予測があった場合には分岐処理に伴うペナルティは発生していない。

【0061】

図4（b）は分岐が成立しない場合の動作である。分岐が成立した場合と比べ、サイクル3で第2のデコーダ302を用いてデコードを行う点が異なっている。

【0062】

まず、サイクル1で制限付条件分岐命令がフェッチされる。ここで制限付条件分岐命令はフェッチ部103に格納されている。さらに制限付条件分岐命令がプリデコードされ、分岐先アドレスが計算される。

【0063】

次に、サイクル2では、分岐するとの予測のもと、分岐先の命令がフェッチ部102にフェッチされる。また、第1のデコーダ301において制限付条件分岐命令がデコードされ、分岐が成立しないことが検知される。

【0064】

次に、サイクル3では、サイクル2のデコードステージで制限付条件分岐命令の分岐が成立しないことが検知されているので、第2のデコーダ302を用いてデコードを行う。記憶装置100から制限付条件分岐命令の次に置かれている命

令が命令レジスタ 3 0 0 へ供給され、当該命令レジスタ 3 0 0 の値とデコーダ選択信号 3 0 3 とが第 2 のデコーダへ供給されデコードされる。ここで、制限付条件分岐命令の次に置かれる命令の種類は制限されており、第 2 のデコーダ 3 0 2 は当該命令のデコードに特化したモジュールであるので、短時間でデコードを行うことができる。さらに、サイクル 3 では制限付条件分岐命令の次の次に置かれた命令がフェッチ部 1 0 2 にフェッチされる。

【 0 0 6 5 】

次に、サイクル 4 では、選択部 3 0 7 および選択部 3 0 5 により第 2 のデコーダ 3 0 2 の出力が選択されて実行ステージに渡される。デコードステージでは、第 1 のデコーダ 3 0 1 が、第 2 のデコーダ 3 0 2 が出力する信号と命令レジスタ 3 0 0 の値を用いてデコードを行う。

【 0 0 6 6 】

なお、図にはないがサイクル 4 の次のサイクル以降、制限付条件分岐命令が現われるまでは、選択部 3 0 7 および選択部 3 0 5 は第 1 のデコーダ 3 0 2 の出力を選択する。

【 0 0 6 7 】

このように、本実施形態 2 のマイクロプロセッサでは、分岐予測が外れた場合においても、分岐処理に伴うペナルティは発生しない。

【 0 0 6 8 】

以上、本発明の実施形態 2 にかかるマイクロプロセッサによれば、分岐予測を行うマイクロプロセッサにおいて、分岐予測が外れた場合でも、次に実行される命令が制限され、当該命令のデコードに特化した第 2 のデコードを用いることにより即座にデコードを実行し、デコード結果を実行ステージに渡すことができるので、ペナルティの発生を防止することができ、処理効率を向上させ、低消費電力化を達成できる。

【 0 0 6 9 】

(実施形態 3)

以下、本発明の命令列変換装置を説明する。

【 0 0 7 0 】

実施形態 3 の命令列変換装置は、実施形態 1 または実施形態 2 に示したマイクロプロセッサにおいて用いる命令列を生成するものである。従来技術のコンパイラによるコンパイル結果を入力し、命令列の中にある条件分岐命令のうち、本発明で用いる制限付条件分岐命令に書き換えられるものを検知し、それらを制限付き条件分岐命令に変換するものである。

【 0 0 7 1 】

本実施形態 3 の命令列変換装置は、分岐する分岐予測を行って動作するマイクロプロセッサに用いる命令列を生成する。そこで、分岐が成立しなかった場合に次に置かれる命令が制限されている命令を制限付条件分岐命令とする。入力された命令列において条件分岐命令を検出した場合、当該条件分岐が成立しなかった場合に次に実行される命令を調べ、当該条件分岐命令と分岐が成立しなかった場合に次に実行される命令との関係が、制限付き条件分岐命令と分岐が成立しなかった場合に次に実行できる命令との関係である場合、当該条件分岐命令を制限付条件分岐命令で置き換える。

【 0 0 7 2 】

図 5 は実施形態 3 の命令列変換装置における命令列変換処理ステップを示したフローチャートである。

【 0 0 7 3 】

図 5 において、500 は入力されたコンパイル結果からアセンブラ命令を 1 つずつ抽出する命令抽出ステップ、501 は命令を全て抽出し終えたか判断する抽出完了判定ステップ、502 は抽出した命令が条件分岐命令か否か判断する条件分岐命令抽出ステップ、503 は条件分岐命令が抽出された場合、次の命令を抽出する制限命令抽出ステップ、504 は制限命令抽出ステップ 503 で抽出された命令が制限付条件分岐命令かどうか判定する制限判定ステップ、505 は制限判定ステップ 504 で制限命令抽出ステップ 503 で抽出された命令が制限を満たすと判断された場合、命令抽出ステップ 500 で抽出されていた条件分岐命令を制限付条件分岐命令で置き換える制限付条件分岐命令変換ステップ、506 は命令列変換処理の終了ステップである。

【 0 0 7 4 】

次に、命令列変換装置における命令列変換処理の実例を挙げて説明する。

【0075】

例えば、制限付条件分岐命令において分岐しない場合に次に置かれた命令が比較命令 `cmp` であれば制限を満たした命令とする。いま、図 6 (a) に示した `case` 文によるプログラムがあり、当該プログラムを従来技術におけるコンパイラでコンパイルされ、図 6 (b) に示したアセンブラ記述されたコンパイル結果である命令列が得られたとする。本発明の命令列変換装置に当該アセンブラ記述された命令列を入力すると、命令抽出ステップ 500、抽出完了判定ステップ 501 によりアセンブラ命令が 1 つずつ抽出される。条件分岐命令抽出ステップ 502 により抽出命令が条件分岐命令 `jz` か否か判断され、制限命令抽出ステップ 503 および制限判定ステップ 504 において抽出された後続命令が比較命令 `cmp` か否か判断される。後続命令が比較命令 `cmp` であると判断されれば、制限付条件分岐命令変換ステップ 505 において条件分岐命令 `jz` が制限付条件分岐命令 `cjz` で置き換える。3 行目、5 行目、7 行目の条件分岐命令 `jz` から制限付条件分岐命令 `cjz` に置き換えられる。このように命令列が変換された結果、図 6 (c) に示した変換後の命令列が得られる。

【0076】

以上、条件分岐命令の次に配置された命令が限定された命令である場合に、当該条件分岐命令を制限付条件分岐命令に置き換えることにより命令列を変換することができる。つまり、従来技術のコンパイラでコンパイルされた命令列を変換して実施形態 1 や実施形態 2 で示したマイクロプロセッサで利用することができる。

【0077】

(実施形態 4)

以下、本発明の実施形態 4 のマイクロプロセッサを説明する。

【0078】

実施形態 4 のマイクロプロセッサは、分岐が成立しないとの分岐予測を行い、分岐が成立し、分岐予測が外れた場合でもペナルティ発生を防止するものである。

【0079】

制限付条件分岐命令を用いる点は実施形態1と同様であるが、ここでいう制限付条件分岐命令とは、条件分岐命令の一種であり、分岐が成立した場合次に実行される分岐先命令が制限されているという命令である。制限付条件分岐命令のデコードの結果、分岐が成立することを検出した場合、次に実行される分岐先命令を通常命令のフェッチとデコードに要するサイクル数よりも少ないサイクル数でフェッチとデコードを行うことによりペナルティ発生を防止する。

【0080】

本実施形態4では、分岐先命令を少ないサイクル数でフェッチとデコードを行う手段として、命令列を格納する記憶装置と分岐先命令のオペコードを格納する特殊レジスタとを備え、制限付条件分岐命令をデコードした結果、分岐することを検出したとき、オペコードを特殊レジスタから、オペランドを記憶装置からデコーダへ供給するという手段を用いる。

【0081】

なお、分岐先に配置された命令を限定された命令とすることで、実施形態1や実施形態2のマイクロプロセッサの動作においてはサイクル1で制限付条件分岐命令をプリデコードして分岐先アドレスを計算する必要があるが、本実施形態4および次の実施形態5ではこのようなプリデコードは必要ない。

【0082】

上記処理を実現する構成例として命令フェッチ部と記憶装置と命令レジスタを模式的に示した構成図は、図1と同様であるのでここでは説明を省略する。なお、上記したように、本実施形態4の構成では、特殊レジスタ101には分岐先命令のオペコードが格納されている。

【0083】

次に、図7は、実施形態4のマイクロプロセッサの動作を示したタイミングチャートである。

【0084】

図7において、縦軸のIF, DEC, EX, ME, WBはそれぞれ5段パイプラインのそれぞれのステージを意味し、図2と同様である。また、図中、WBの

下には命令レジスタ104に供給されるオペコード部とオペランド部の供給元が示されている。

【0085】

図7(a)は分岐が成立しないとの分岐予測が外れ、分岐が成立する場合の動作である。

【0086】

サイクル1で制限付条件分岐命令がフェッチされる。ここで制限付条件分岐命令はフェッチ部103に格納されている。

【0087】

次に、サイクル2では、分岐しないとの予測のもと、フェッチ部102には制限付条件分岐命令の次に置かれている命令がフェッチ部102にフェッチされる。一方、DECステージでは、制限付条件分岐命令がデコードされ、分岐が成立することが検知される。さらに、分岐先アドレスが計算される。

【0088】

次に、サイクル3では、分岐先命令のオペコードが特殊レジスタ101から供給され、オペランドが記憶装置100から読み出された値が命令レジスタ104へ供給される。ここで、分岐先命令のオペコードは特殊レジスタ101に格納されたオペコードと一致するように命令の種類が制限されているため、特殊レジスタ101からオペコードを命令レジスタ104に供給することで分岐先命令と同等のオペコードのデコードを早く開始できる。また、オペランドのデコードはレジスタを選択するためだけのものなので、記憶装置100から供給してもサイクル3内でデコードできる。

【0089】

なお、サイクル3のフェッチステージでは、記憶装置100から分岐先の命令を読み出すと同時にその次に配置された命令がフェッチ部102にフェッチされる。

【0090】

次に、サイクル4では分岐先命令の次に置かれている命令がデコードされ、分岐先命令の次の次に置かれている命令がフェッチ部102にフェッチされる。

【 0 0 9 1 】

このように、本実施形態 1 のマイクロプロセッサでは、分岐しないとの分岐予測が外れた場合においても、分岐処理に伴うペナルティは発生しない。

【 0 0 9 2 】

図 7 (b) は、分岐が成立しないとの分岐予測が当たった場合の動作である。

【 0 0 9 3 】

サイクル 1 で制限付条件分岐命令がフェッチされる。ここで制限付条件分岐命令はフェッチ部 1 0 3 に格納されている。さらに制限付条件分岐命令がプリデコードされ、分岐先アドレスが計算される。次に、サイクル 2 では、分岐しないとの予測のもと、制限付条件分岐命令の次に置かれた命令がフェッチ部 1 0 2 にフェッチされる。また、DEC ステージでは、制限付条件分岐命令がデコードされ、分岐が成立しないことが検知される。さらに、分岐先アドレスが計算される。次に、サイクル 3 では、制限付条件分岐命令の次の命令がフェッチ部 1 0 2 にフェッチされ、制限付条件分岐命令の次の命令は DEC ステージにおいてデコードされる。このように分岐しないとの分岐予測が当たった場合には分岐処理に伴うペナルティは発生していない。

【 0 0 9 4 】

以上、本発明の実施形態 4 にかかるマイクロプロセッサによれば、分岐が成立しないとの分岐予測を行うマイクロプロセッサにおいて、分岐予測が外れた場合でも、分岐先命令のオペコードが特殊レジスタから、当該命令のオペランドが記憶装置からデコーダへ即座に供給され、デコード結果を実行ステージに渡すことができるので、ペナルティの発生を防止することができ、処理効率を向上させ、低消費電力化を達成できる。

【 0 0 9 5 】

(実施形態 5)

以下、本発明の実施形態 5 のマイクロプロセッサを説明する。

【 0 0 9 6 】

実施形態 5 のマイクロプロセッサも、実施形態 4 と同様、分岐が成立しないとの分岐予測を行い、分岐が成立した場合でもペナルティ発生を防止するものであ

る。また、制限付条件分岐命令を用いる点においても実施形態 4 と同様である。

【 0 0 9 7 】

本実施形態 5 では、分岐しないとの分岐予測が外れた場合に分岐先命令を少ないサイクル数でフェッチとデコードを行う手段として、通常の命令列のデコードに用いる第 1 のデコーダと、分岐先命令のデコードに特化してその命令を通常のデコードサイクル数よりも短いサイクル数でデコードする第 2 のデコーダを用いる。通常、第 1 のデコーダを用い、第 1 のデコーダが制限付条件分岐命令をデコードし、分岐することを検出したときは、分岐先命令のデコードサイクルにおいて第 2 のデコーダを用いて短時間でデコードを実行するものである。

【 0 0 9 8 】

上記処理を実現する構成例として命令フェッチ部と記憶装置と命令レジスタを模式的に示した構成図は、図 3 と同様であるのでここでは説明を省略する。

【 0 0 9 9 】

なお、本実施形態 5 では、第 1 のデコーダが D E C ステージで通常メインに用いられる構成であり、第 2 のデコーダ 3 0 2 は分岐先命令のデコードに特化したモジュールでありハードウェア規模が小さくデコードの時間が短い。分岐先命令を通常のデコードサイクル数よりも短いサイクル数でデコードするものである。

【 0 1 0 0 】

次に、図 8 は、実施形態 5 のマイクロプロセッサの動作を示したタイミングチャートである。

【 0 1 0 1 】

図 8 において、縦軸の I F, D E C, E X, M E, W B はそれぞれ 5 段パイプラインのそれぞれのステージを意味し、図 4 と同様である。また、図中、W B の下には選択されるデコーダが示されている。

【 0 1 0 2 】

図 8 (a) は分岐が成立しないとの分岐予測が外れ、分岐が成立する場合の動作である。

【 0 1 0 3 】

まず、サイクル 1 で制限付条件分岐命令がフェッチされる。ここで制限付条件

分岐命令はフェッチ部 1 0 3 に格納されている。

【 0 1 0 4 】

次に、サイクル 2 では、分岐しないとの予測のもと、制限付条件分岐命令の次に置かれた命令がフェッチ部 1 0 2 にフェッチされる。また、第 1 のデコーダ 3 0 1 において制限付条件分岐命令がデコードされ、分岐が成立することが検知される。さらに、分岐先アドレスが計算される。

【 0 1 0 5 】

次に、サイクル 3 では、サイクル 2 のデコードステージで制限付条件分岐命令の分岐が成立しないことが検知されているので、第 2 のデコーダ 3 0 2 を用いてデコードを行う。記憶装置 1 0 0 から分岐先命令が命令レジスタ 3 0 0 へ供給され、当該命令レジスタ 3 0 0 の値とデコーダ選択信号 3 0 3 とが第 2 のデコーダへ供給されデコードされる。ここで、分岐先命令の種類は制限されており、第 2 のデコーダ 3 0 2 は当該命令のデコードに特化したモジュールであるので、短時間でデコードを行うことができる。さらに、サイクル 3 では分岐先命令の次の次に置かれた命令がフェッチ部 1 0 2 にフェッチされる。

【 0 1 0 6 】

次に、サイクル 4 では、選択部 3 0 7 および選択部 3 0 5 により第 2 のデコーダ 3 0 2 の出力が選択されて実行ステージに渡され、デコードステージでは分岐先命令の次に置かれた命令がデコードされ、第 1 のデコーダ 3 0 1 は、第 2 のデコーダ 3 0 2 が出力する信号と命令レジスタ 3 0 0 の値を用いてデコードを行う。また、フェッチステージでは分岐先命令の次の次に置かれた命令がフェッチ部 1 0 3 にフェッチされる。

【 0 1 0 7 】

なお、図にはないがサイクル 4 の次のサイクル以降、制限付条件分岐命令が現われるまでは、選択部 3 0 7 および選択部 3 0 5 は第 1 のデコーダ 3 0 2 の出力を選択する。

【 0 1 0 8 】

このように、本実施形態 5 のマイクロプロセッサでは、分岐するとの分岐予測が外れた場合においても、分岐処理に伴うペナルティは発生しない。

【0109】

図8(b)は分岐予測が当たり、分岐が成立しない場合の動作である。ここではデコードは全て第1のデコーダ301で行う。まず、サイクル1で制限付条件分岐命令がフェッチされる。ここで制限付条件分岐命令はフェッチ部103に格納されている。次に、サイクル2ではフェッチステージでは分岐しないとの予測のもと、制限付条件分岐命令の次に置かれた命令がフェッチ部102にフェッチされる。一方、第1のデコーダ301では、制限付条件分岐命令がデコードされ、分岐が成立することが検知される。さらに、分岐先アドレスが計算される。次に、サイクル3では制限付条件分岐命令の次の次の命令がフェッチ部102にフェッチされ、制限付条件分岐命令の次の命令は第1のデコーダ301においてデコードされる。このように分岐しないとの分岐予測があった場合には分岐処理に伴うペナルティは発生していない。

【0110】

以上、本発明の実施形態5にかかるマイクロプロセッサによれば、分岐が成立しないとの分岐予測を行うマイクロプロセッサにおいて、分岐予測が外れた場合でも、分岐先命令が制限され、当該命令のデコードに特化した第2のデコードを用いることにより即座にデコードを実行し、デコード結果を実行ステージに渡すことができるので、ペナルティの発生を防止することができ、処理効率を向上させ、低消費電力化を達成できる。

【0111】

(実施形態6)

以下、本発明の実施形態6にかかる命令列変換装置を説明する。

【0112】

実施形態6の命令列変換装置は、実施形態4または実施形態5に示したマイクロプロセッサにおいて用いる命令列を生成するものである。従来技術のコンパイラによるコンパイル結果を入力し、命令列の中にある条件分岐命令のうち、本発明で用いる制限付条件分岐命令に書き換えられるものを検知し、それらを制限付き条件分岐命令に変換するものである。

【0113】

本実施形態 6 の命令列変換装置は、分岐しないとの分岐予測を行って動作するマイクロプロセッサに用いる命令列を生成する。そこで、分岐が成立した場合に分岐先命令が制限されている命令を制限付条件分岐命令とする。入力された命令列において条件分岐命令を検出した場合、当該条件分岐が成立した場合に分岐先命令を調べ、当該条件分岐命令と分岐先命令との関係が、制限付き条件分岐命令と分岐先命令との関係である場合、当該条件分岐命令を制限付条件分岐命令で置き換える。

【 0 1 1 4 】

図 9 は実施形態 6 の命令列変換装置における命令列変換処理ステップを示したフローチャートである。

【 0 1 1 5 】

図 9 において、9 0 0 は入力されたコンパイル結果からアセンブラ命令を 1 つずつ抽出する命令抽出ステップ、9 0 1 は命令を全て抽出し終えたか判断する抽出完了判定ステップ、9 0 2 は抽出した命令が条件分岐命令か否か判断する条件分岐命令抽出ステップ、9 0 3 は条件分岐命令が抽出された場合、分岐先の命令を抽出する制限命令抽出ステップ、9 0 4 は制限命令抽出ステップ 9 0 3 で抽出された分岐先命令が制限付条件分岐命令たる制限条件を満たすかどうか判定する制限判定ステップ、9 0 5 は制限判定ステップ 9 0 4 で制限命令抽出ステップ 9 0 3 で抽出された命令が制限を満たすと判断された場合、命令抽出ステップ 9 0 0 で抽出されていた条件分岐命令を制限付条件分岐命令で置き換える制限付条件分岐命令変換ステップ、9 0 6 は命令列変換処理の終了ステップである。

【 0 1 1 6 】

次に、命令列変換装置における命令列変換処理の実例を挙げて説明する。

【 0 1 1 7 】

例えば、制限付条件分岐命令において分岐する場合に分岐先命令が比較命令 `cmp` であるとの制限条件を設ける。いま、図 1 0 (a) に示した `case` 文によるプログラムがあり、当該プログラムを従来技術におけるコンパイラでコンパイルされ、図 1 0 (b) に示したアセンブラ記述されたコンパイル結果である命令列が得られたとする。なお、図 1 0 (a) の `case` 文は図 6 (a) の `case`

文とまったく同じものであるが、コンパイラの変換規則の違いからコンパイル結果が異なったものとなっている。図 6 (b) では、比較の結果値が一致していた場合に分岐する条件分岐命令 `j z` を用いているが、図 1 0 (b) では比較の結果値が一致していなかった場合に分岐する条件分岐命令 `j n z` を用いている。このため、図 6 (b) では条件分岐命令 `j z` の直後に比較命令 `c m p` が置かれることが多いのに対して、図 1 0 (b) では、条件分岐命令 `j n z` の分岐先に比較命令 `c m p` が置かれることが多くなっている。図 1 0 (b) では条件分岐命令の分岐先命令が特定の比較命令 `c m p` に制限されている場合を扱う。

【 0 1 1 8 】

本実施形態 6 の命令列変換装置に当該アセンブラ記述された命令列を入力すると、命令抽出ステップ 9 0 0、抽出完了判定ステップ 9 0 1 によりアセンブラ命令が 1 つずつ抽出される。条件分岐命令抽出ステップ 9 0 2 により抽出命令が条件分岐命令 `j n z` か否か判断され、制限命令抽出ステップ 9 0 3 および制限判定ステップ 9 0 4 において抽出された分岐先命令が比較命令 `c m p` か否か判断される。分岐先命令が比較命令 `c m p` であると判断されれば、制限付条件分岐命令変換ステップ 9 0 5 において条件分岐命令 `j n z` が制限付条件分岐命令 `c j n z` で置き換える。3 行目、9 行目、1 4 行目の条件分岐命令 `j n z` から制限付条件分岐命令 `c j n z` に置き換えられる。このように命令列が変換された結果、図 1 0 (c) に示した変換後の命令列が得られる。

【 0 1 1 9 】

以上、条件分岐命令の分岐先命令が限定された命令である場合に、当該条件分岐命令を制限付条件分岐命令に置き換えることにより命令列を変換することができる。つまり、従来技術のコンパイラでコンパイルされた命令列を変換して実施形態 4 や実施形態 5 で示したマイクロプロセッサで利用することができる。

【 0 1 2 0 】

なお、以上に示した実施形態 1 と実施形態 4 のマイクロプロセッサでは、特殊レジスタに固定的に特定の 1 命令（例えば比較命令 `c m p`）を格納しておく場合を述べたが、`c a s e` 文毎に特殊レジスタの値を設定し直しても良い。また、オペコードの長さが一定である場合を想定して述べたが、特殊レジスタにオペコー

ドの長さを定義する領域を設けて可変にできるようにしても良い。

【 0 1 2 1 】

さらに、以上に示した実施形態 2 と実施形態 5 では、固定的に特定の 1 命令（例えば比較命令 `cmp`）に特化してデコードする専用の第 2 のデコーダを想定して述べたが、複数命令を高速にデコードできるものであっても良い。

【 0 1 2 2 】

【発明の効果】

本発明に係るマイクロプロセッサは、次に実行される命令が制限されている制限付条件分岐命令を用いるので、分岐が成立するとの分岐予測を行って分岐予測が外れた場合でも、通常命令のフェッチとデコードに要するサイクル数よりも少ないサイクル数でフェッチとデコードを行って即座に実行ステージに渡すことができるのでペナルティの発生を防止することができ、処理効率を向上させ、低消費電力化を達成できる。

【 0 1 2 3 】

また、本発明に係るマイクロプロセッサは、分岐先命令が制限されている制限付条件分岐命令を用いるので、分岐が成立しないとの分岐予測を行って分岐予測が外れた場合でも、通常命令のフェッチとデコードに要するサイクル数よりも少ないサイクル数でフェッチとデコードを行って即座に実行ステージに渡すことができるのでペナルティの発生を防止することができ、処理効率を向上させ、低消費電力化を達成できる。

【 0 1 2 4 】

なお、本発明に係る命令列変換装置は、従来技術において用いられる通常のコンパイラによるプログラムのコンパイル結果を入力として条件付分岐命令を制限付条件分岐命令列に置き換えることができ、上記マイクロプロセッサに用いる制限付条件分岐命令を含む命令列が供給される。

【図面の簡単な説明】

【図 1】 本発明の実施形態 1 における命令フェッチ部と記憶装置と命令レジスタの構成を模式的に示した図

【図 2】 本発明の実施形態 1 のマイクロプロセッサの動作を示したタイミン

グチャート

【図 3】 本発明の実施形態 2 におけるデコーダを中心とした構成を模式的に示した図

【図 4】 本発明の実施形態 2 の構成の動作を示したタイミングチャート

【図 5】 本発明の実施形態 3 の命令列変換装置における命令列変換処理ステップを示したフローチャート

【図 6】 (a) は c a s e 文によるプログラムの例を示す図、(b) は (a) をアセンブラ記述したコンパイル結果の例を示す図、(c) は変換後の命令列の例を示す図

【図 7】 本発明の実施形態 4 のマイクロプロセッサの動作を示したタイミングチャート

【図 8】 本発明の実施形態 5 のマイクロプロセッサの動作を示したタイミングチャート

【図 9】 実施形態 6 の命令列変換装置における命令列変換処理ステップを示したフローチャート

【図 1 0】 (a) は c a s e 文によるプログラムの例を示す図、(b) は (a) をアセンブラ記述したコンパイル結果の例を示す図、(c) は変換後の命令列の例を示す図

【図 1 1】 従来技術における分岐予測機能を備えたマイクロプロセッサの動作を示したタイミングチャート

【図 1 2】 従来技術における遅延分岐機能を備えたマイクロプロセッサの動作を示したタイミングチャート

【図 1 3】 (a) は c a s e 文によるプログラムの例を示す図、(b) は (a) をアセンブラ記述したコンパイル結果の例を示す図

【符号の説明】

1 0 0 記憶装置

1 0 1 特殊レジスタ

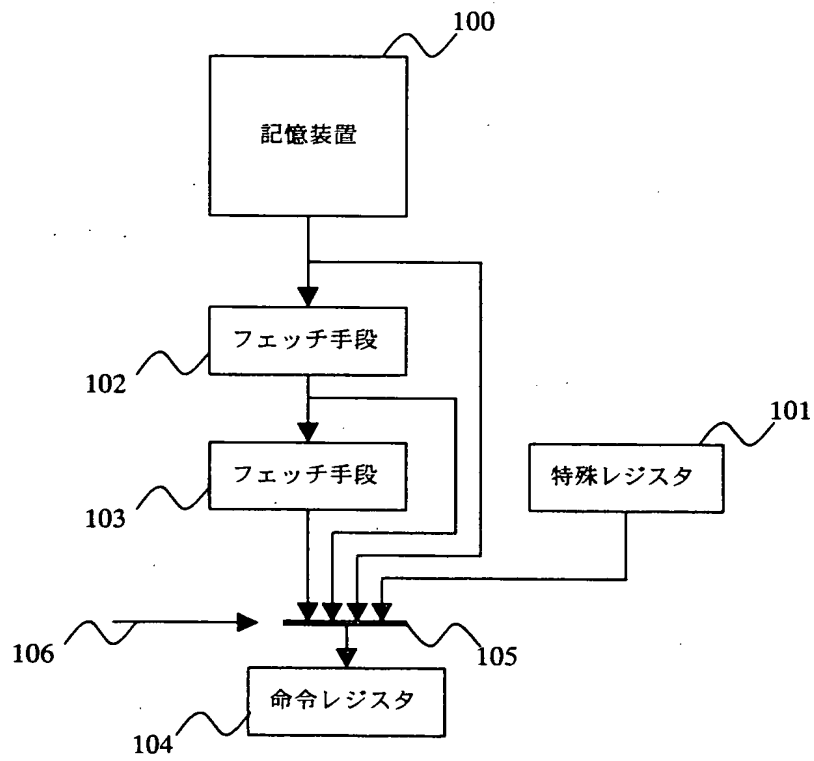
1 0 2、1 0 3 フェッチ部

1 0 4 命令レジスタ

- 1 0 5 選択部
- 1 0 6 選択制御信号
- 3 0 0 命令レジスタ
- 3 0 1 第 1 のデコーダ
- 3 0 2 第 2 のデコーダ
- 3 0 3 デコーダ選択信号
- 3 0 4 制御信号
- 3 0 5、3 0 7 選択部
- 3 0 6、3 0 8 選択制御信号
- 5 0 0、1 2 0 0 命令抽出ステップ
- 5 0 1、1 2 0 1 抽出完了判定ステップ
- 5 0 2、1 2 0 2 条件分岐命令抽出ステップ
- 5 0 3、1 2 0 3 制限命令抽出ステップ
- 5 0 4、1 2 0 4 制限判定ステップ
- 5 0 5、1 2 0 5 制限付条件分岐命令変換ステップ
- 5 0 6、1 2 0 6 命令変換部の終了

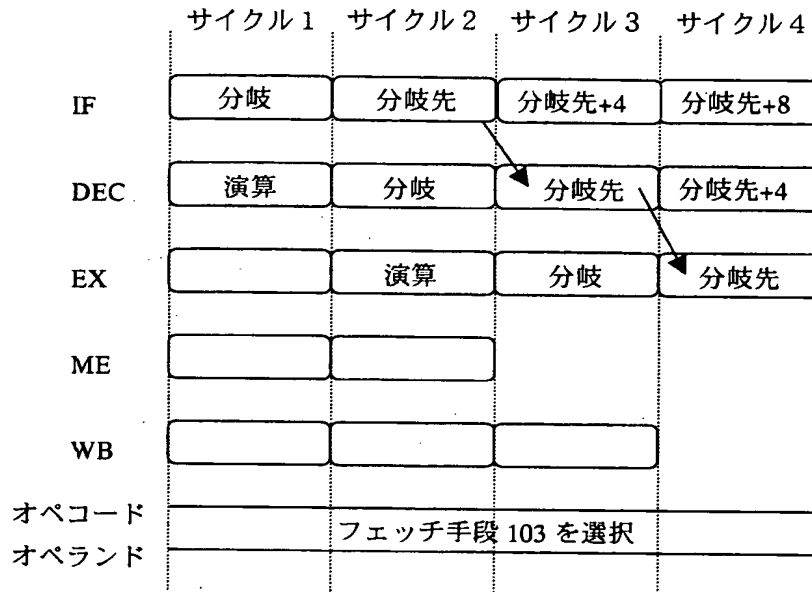
【書類名】 図面

【図 1】

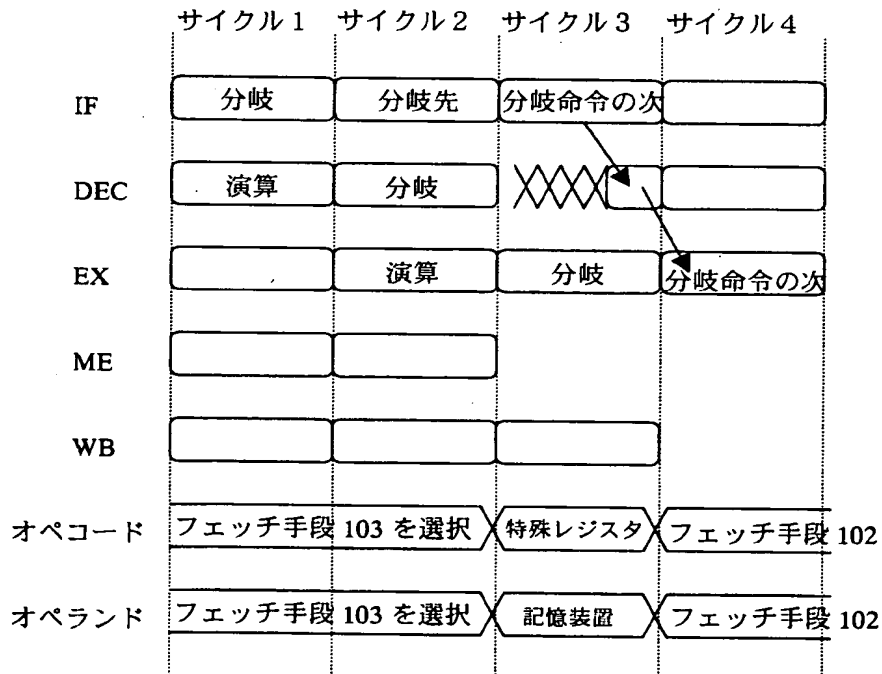


【図 2】

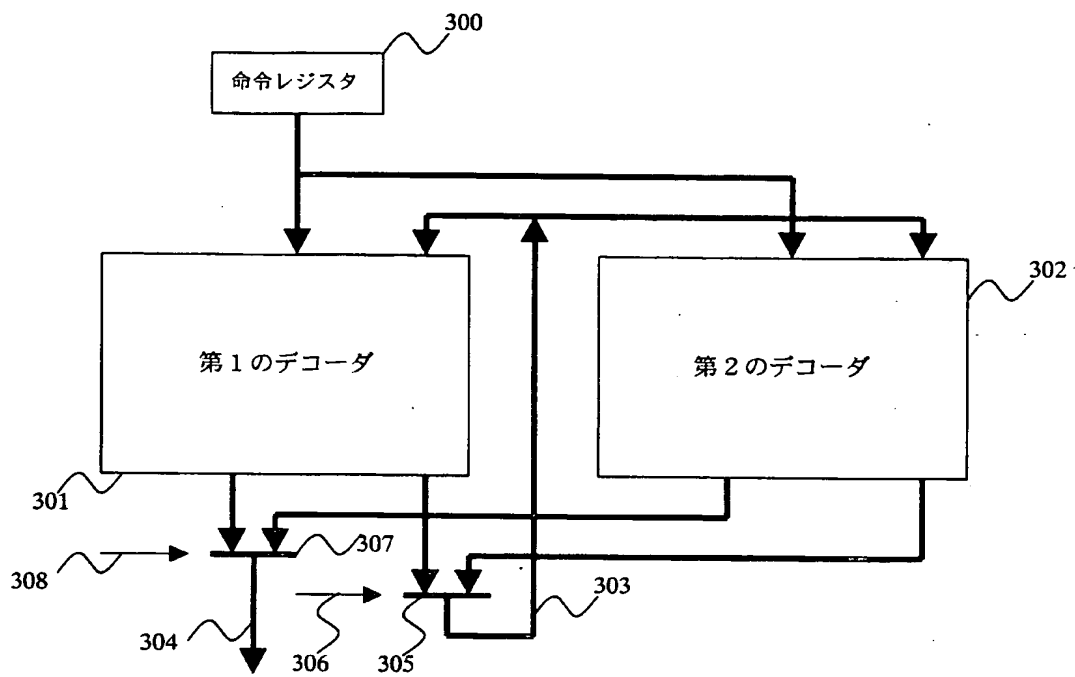
(a)分岐成立



(b)分岐不成立

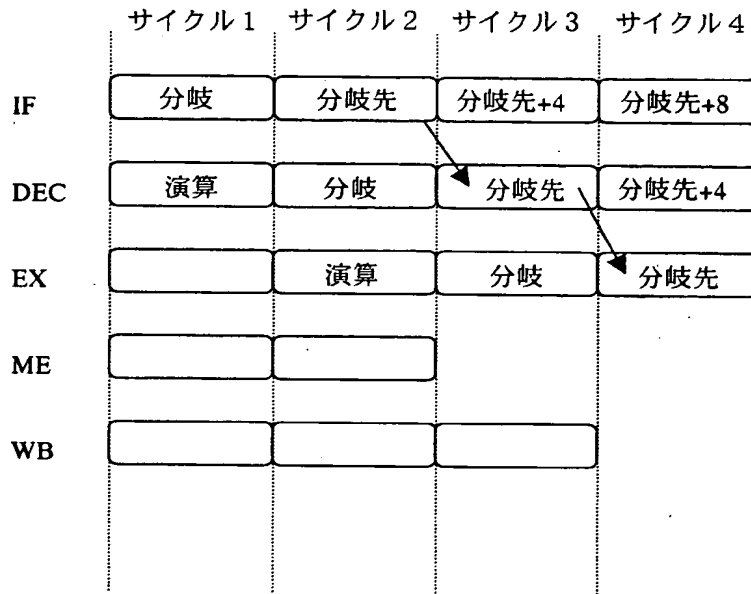


【図 3】

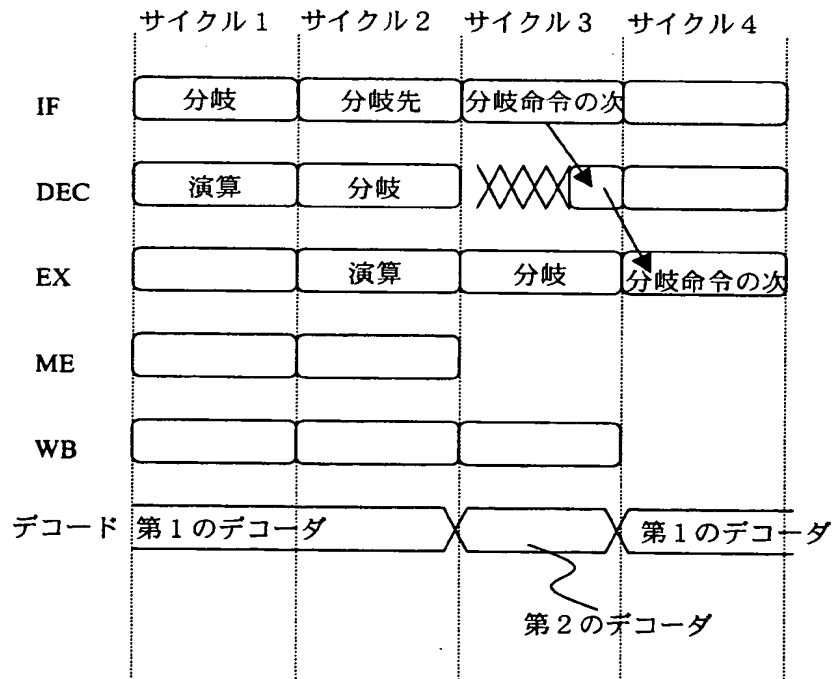


【図 4】

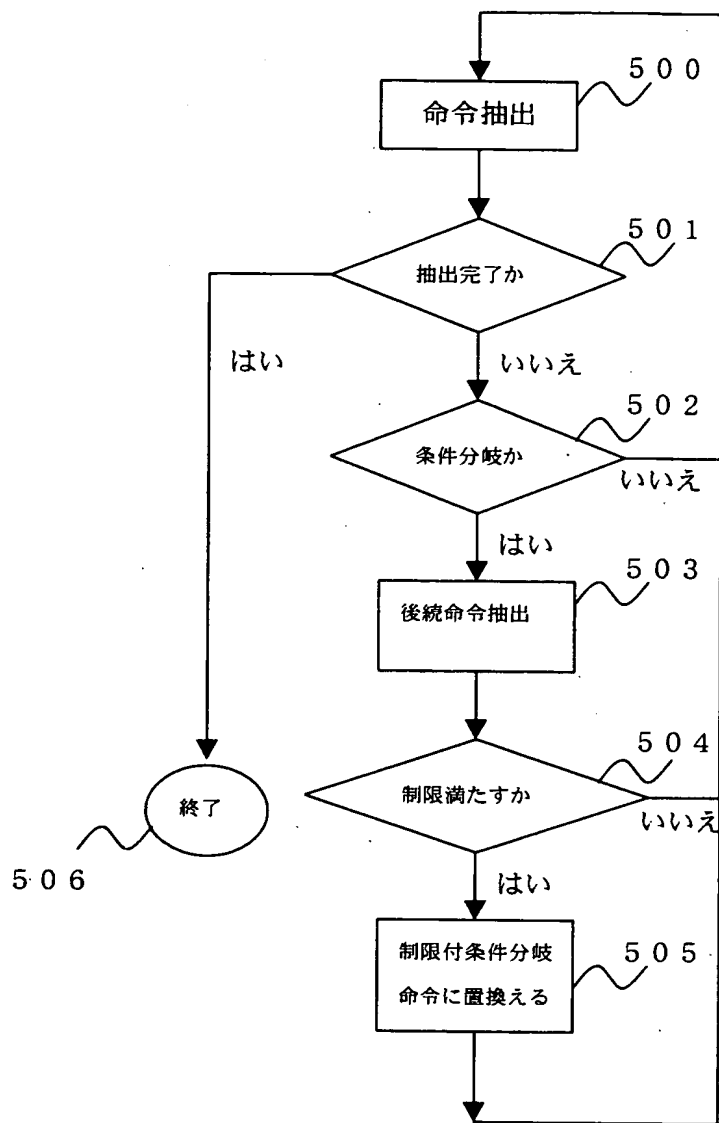
(a) 分岐成立



(b) 分岐不成立



【図 5】



【図 6】

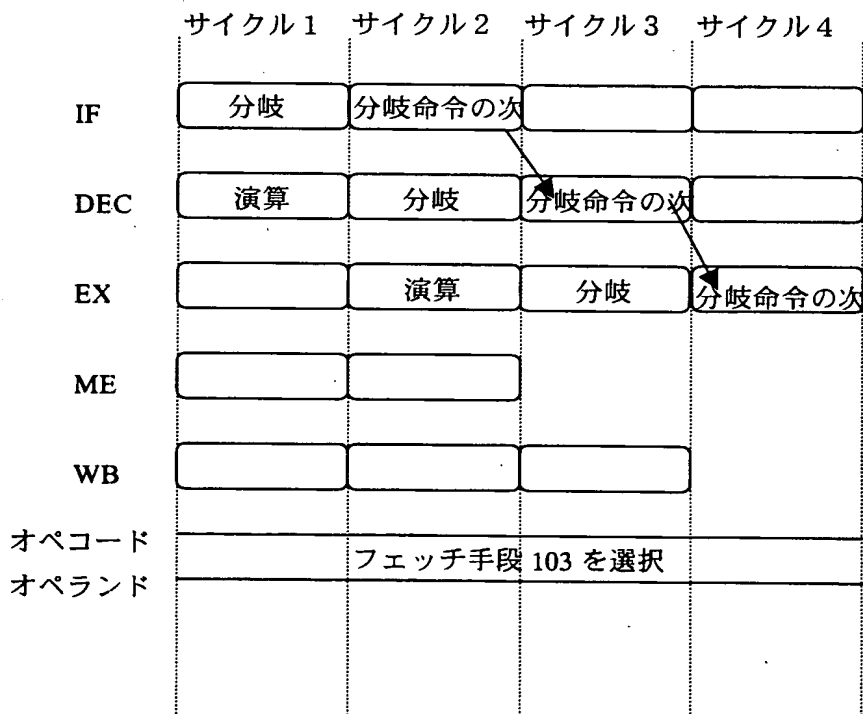
(a) case 文の例	(b) アセンブラ記述	(c) 命令列変換結果
switch(mode)	ld d0,(mode)	ld d0,(mode)
case: 1 a=b+1;	cmp d0,1	cmp d0,1
case: 3 a=c;	jz	cjz
case: 5 a=b+c;	cmp d0,3	cmp d0,3
:	jz	cjz
:	cmp d0,5	cmp d0,5
	jz	cjz
	:	:
	:	:

【図 7】

(a) 分岐成立

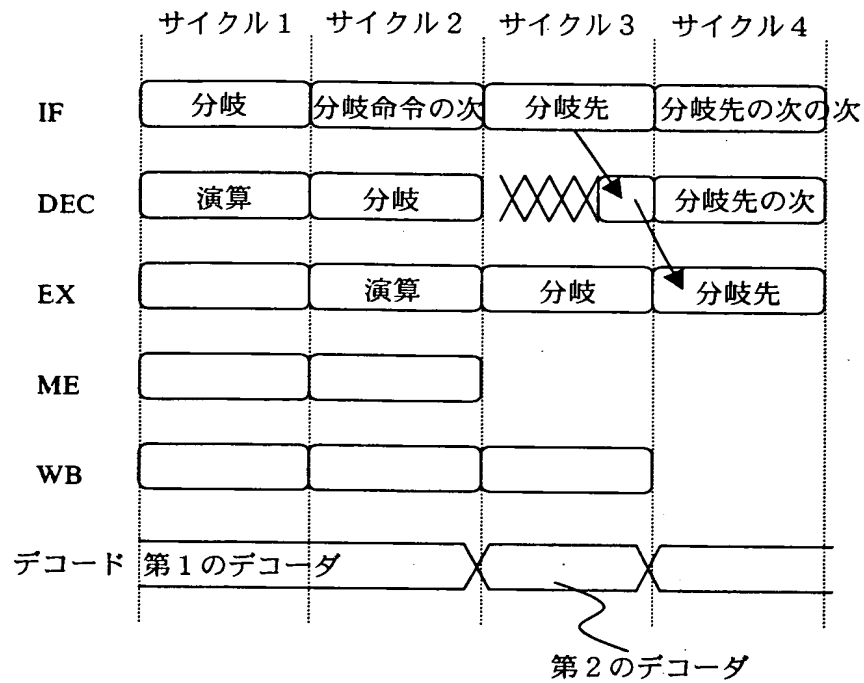


(b) 分岐不成立

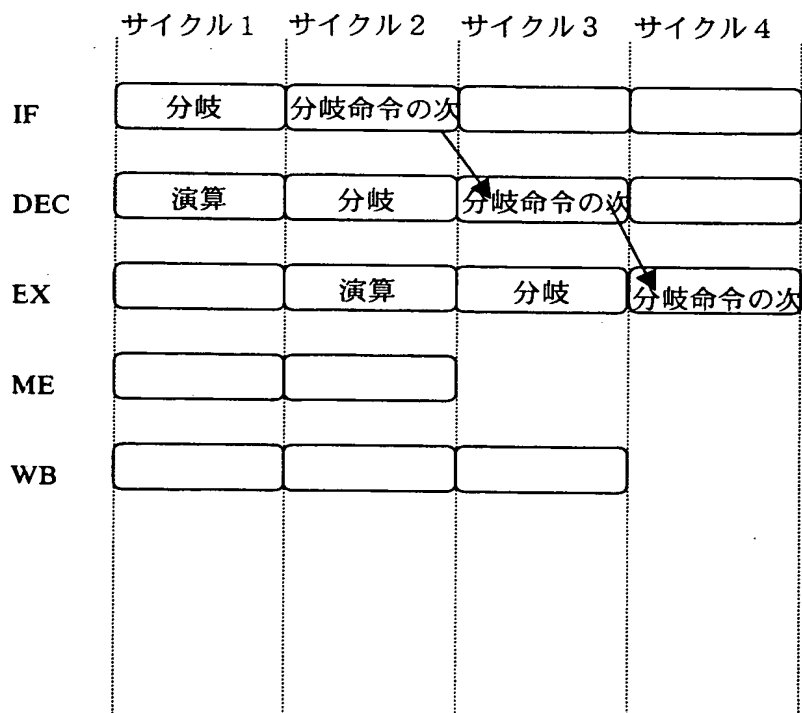


【図 8】

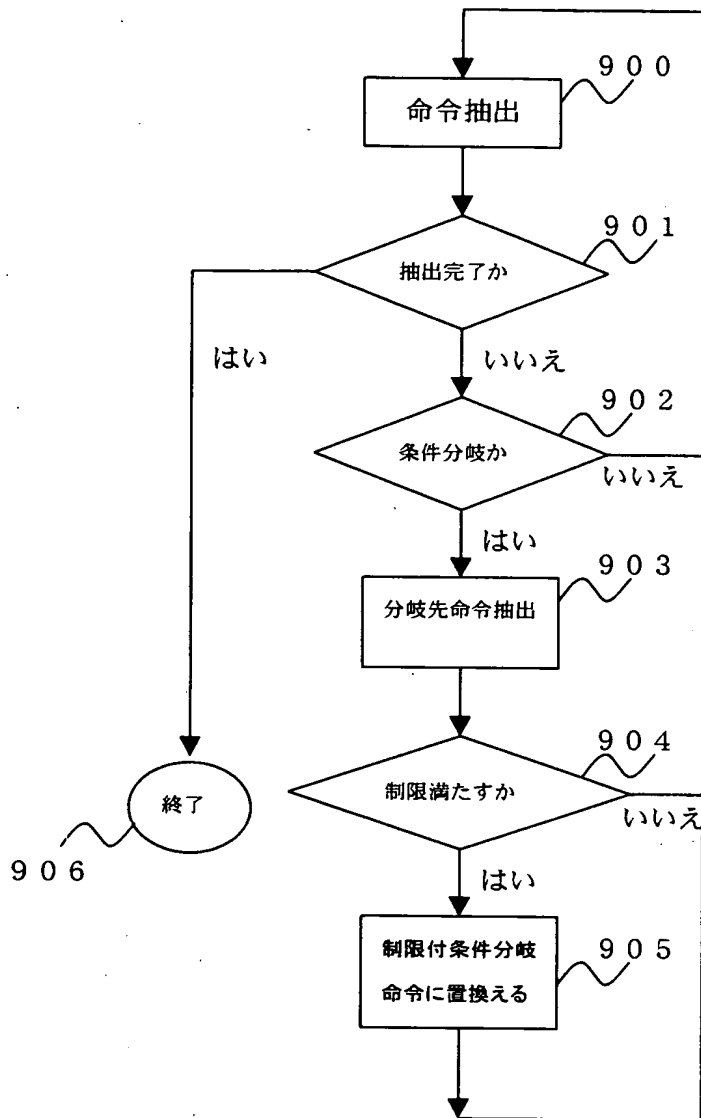
(a) 分岐成立



(b) 分岐不成立



【図9】



【図 1 0】

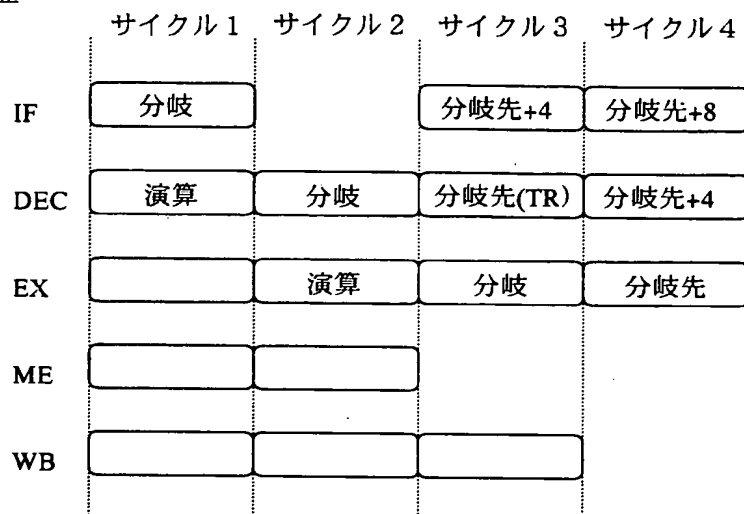
(a) case 文の例
 switch(mode)
 case: 1 a=b+1;
 case: 3 a=c;
 case: 5 a=b+c;
 :
 :

(b) アセンブラ記述
 ld d0,(mode)
 cmp d0,1
 jnz A
 ld d0,(b)
 add d0,1
 st d0,(a)
 jmp END
 A: cmp d0,3
 jnz B
 ld d0,(c)
 st d0,(a)
 jmp END
 B: cmp d0,5
 jnz C
 ld d0,(c)
 ld d1,(b)
 add d0,d1
 st d0,(a)
 :
 :
 END:

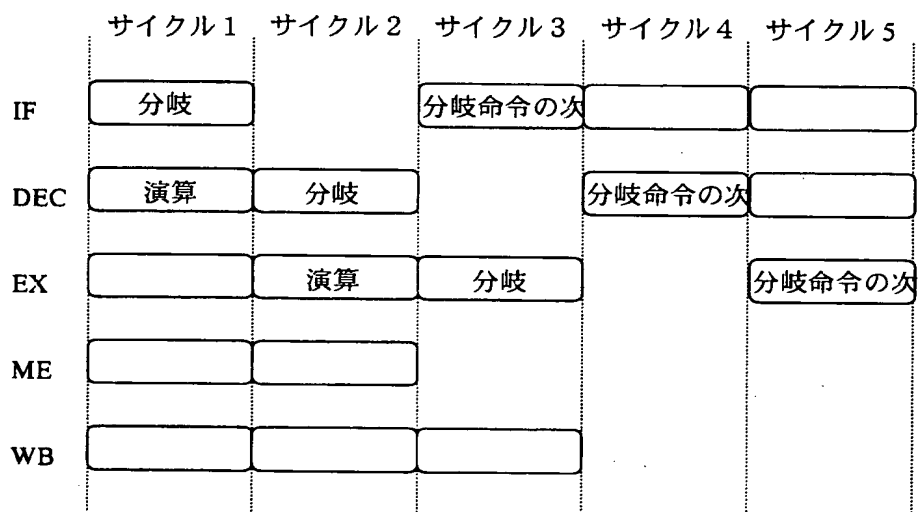
(c) 変換後の命令列
 ld d0,(mode)
 cmp d0,1
 cjnz A
 ld d0,(b)
 add d0,1
 st d0,(a)
 jmp END
 A: cmp d0,3
 cjnz B
 ld d0,(c)
 st d0,(a)
 jmp END
 B: cmp d0,5
 cjnz C
 ld d0,(c)
 ld d1,(b)
 add d0,d1
 st d0,(a)
 :
 :
 END:

【図 1 1】

(a)分岐成立



(b)分岐不成立



【図 1 2】

(a)分岐成立

	サイクル 1	サイクル 2	サイクル 3	サイクル 4
IF	分岐	遅延スロット	分岐先	
DEC	演算	分岐	遅延スロット	分岐先
EX		演算	分岐	遅延スロット
ME				
WB				

(b)分岐不成立

	サイクル 1	サイクル 2	サイクル 3	サイクル 4
IF	分岐	遅延スロット	分岐命令の次	
DEC	演算	分岐	遅延スロット	分岐命令の次
EX		演算	分岐	遅延スロット
ME				
WB				

【図 1 3】

(a) case 文の例	(b) アセンブラ記述
switch(mode)	ld d0,(mode)
case: 1 a=b+1;	cmp d0,1
case: 3 a=c;	jz
case: 5 a=b+c;	cmp d0,3
:	jz
:	cmp d0,5
	jz
	:
	:

【書類名】 要約書

【要約】

【課題】 分岐命令の分岐予測が外れた場合でもペナルティを発生しないようにしてマイクロプロセッサの処理効率低下を防止し、低消費電力を達成する。

【解決手段】 分岐予測が外れた場合でも次に実行される命令が制限されている制限付条件分岐命令を用いる。命令列を格納する記憶装置 1 0 0 と、制限された命令のオペコードを格納する特殊レジスタ 1 0 1 とを備え、制限付条件分岐命令をデコードし、分岐予測が外れたことを検出した場合、次に実行される命令のオペコードを特殊レジスタ 1 0 1 から、オペランドを記憶装置 1 0 0 からデコーダへ供給する。オペコードは特殊レジスタ 1 0 1 からの読み出しによりデコーダに供給でき、デコードが早期に開始できるので短いサイクル数でデコード処理を完了することができ、ペナルティが発生せず、処理効率低下が防止され、低消費電力が達成される。

【選択図】 図 1

認定・付加情報

特許出願の番号	特願 2000-403540
受付番号	50001708313
書類名	特許願
担当官	末武 実 1912
作成日	平成13年 1月12日

<認定情報・付加情報>

【特許出願人】

【識別番号】	000005821
【住所又は居所】	大阪府門真市大字門真1006番地
【氏名又は名称】	松下電器産業株式会社

【代理人】

申請人

【識別番号】	100095555
【住所又は居所】	大阪府大阪市北区西天満4丁目3番25号 梅田 ブラザビル401号室 池内・佐藤特許事務所

【氏名又は名称】	池内 寛幸
----------	-------

【選任した代理人】

【識別番号】	100076576
【住所又は居所】	大阪府大阪市北区西天満4丁目3番25号 梅田 ブラザビル401号室 池内・佐藤特許事務所

【氏名又は名称】	佐藤 公博
----------	-------

【選任した代理人】

【識別番号】	100107641
【住所又は居所】	大阪府大阪市北区西天満4丁目3番25号 梅田 ブラザビル401号室 池内・佐藤特許事務所

【氏名又は名称】	鎌田 耕一
----------	-------

【選任した代理人】

【識別番号】	100110397
【住所又は居所】	大阪府大阪市北区西天満4丁目3番25号 梅田 ブラザビル401号室 池内・佐藤特許事務所

【氏名又は名称】	梶丘 圭司
----------	-------

【選任した代理人】

【識別番号】	100115255
【住所又は居所】	大阪府大阪市北区西天満4丁目3番25号 梅田 ブラザビル401号室 池内・佐藤特許事務所

認定・付加情報（続き）

【氏名又は名称】	辻丸 光一郎
【選任した代理人】	
【識別番号】	100115152
【住所又は居所】	大阪府大阪市北区西天満4丁目3番25号 梅田 プラザビル401号室 池内・佐藤特許事務所
【氏名又は名称】	黒田 茂

出 願 人 履 歴 情 報

識別番号 [0 0 0 0 0 5 8 2 1]

1. 変更年月日 1 9 9 0 年 8 月 2 8 日

[変更理由] 新規登録

住 所 大阪府門真市大字門真 1 0 0 6 番地

氏 名 松下電器産業株式会社